

Geometry Modeling, Flow Domain Modeling, and Grid Generation



Objectives

The objectives of this discussion are to relate experiences and offer some practical advice with regard to:

1. Modeling geometry for use with CFD analysis
2. Modeling the flow domain
3. Generating a grid for the CFD analysis

Geometry modeling, flow domain modeling, and grid generation are often the most difficult and time-intensive aspects of a CFD analysis.

Focus will include multi-zone, structured and unstructured grids.

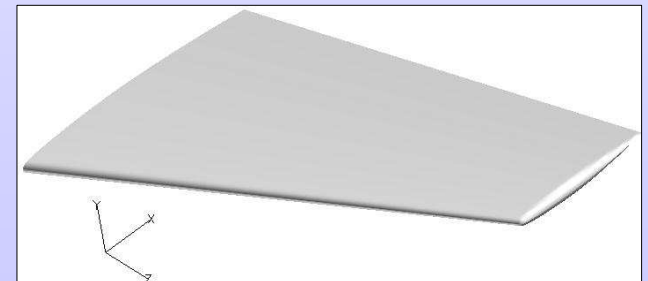
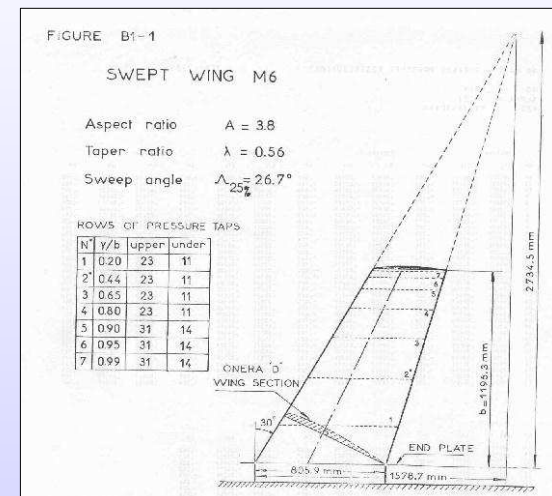
Geometry Data

The data describing the geometry of the body about which the flow is to be computed can be provided in a variety of means:

- Coordinates of points
- Blueprint mechanical drawings
- CAD files

CFD analysis mainly needs the surfaces that “see” the flow.

These surfaces need to be extracted from the geometry data in some manner.



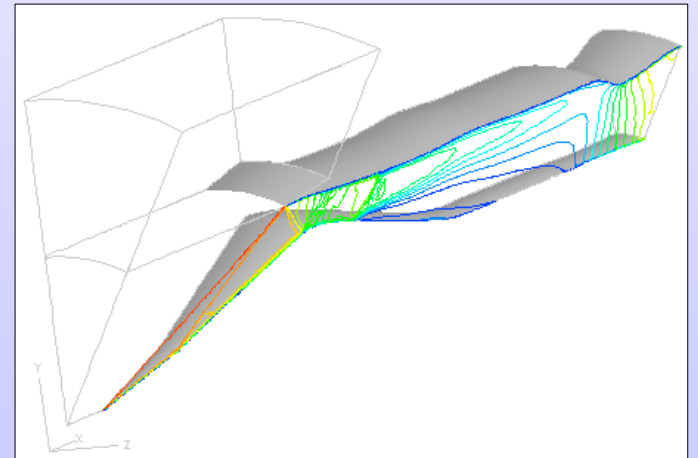
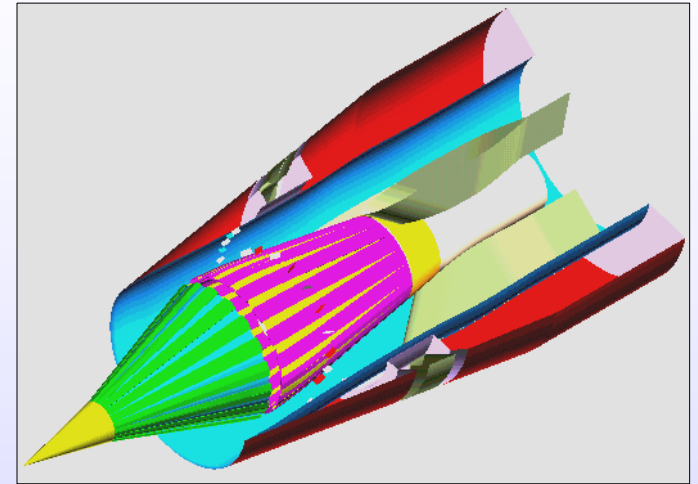
Level of Geometric Detail

Need to be reasonable with the level of geometric detail to be included.

Greater geometric detail leads to more grid points and longer computations.

A sequential approach is best in which the detail of the geometry is increased as the CFD analyses proceed.

For the NASA VDC inlet, neglected VGs, struts, leaves on centerbody, and slot.





CAD Geometry Data

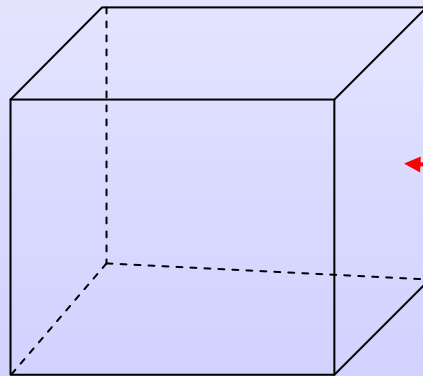
Availability of CAD files for defining the geometry provides needed information in a ready form; however, issues are:

- Geometry data may consist of points, lines, curves, surfaces, and solids.
- CAD files usually contain much more information than is needed by the CFD analysis (mechanical details). Can take considerable effort to extract out the curves and surfaces that are needed.
- Many CAD packages are solid modelers. They can output surfaces; however, the “quilt” of surfaces can have some strange trimmed surfaces that have gaps in their edges.
- CAD files are usually generated by mechanical designers that are mindful of tolerances and clearances needed in the mechanical design; however, CFD desires no clearances or gaps. CFD analyst may need to spend time to “close up” the surfaces.
- CAD file format. IGES file format is commonly used since it does well representing surfaces and curves.
- Mechanical drawings contain all the fine features; however, CFD may not wish to capture such geometric resolution. CFD analyst has to remove such features.

Domain, Zone, Grid, and Cell

The control volumes exists at several levels:

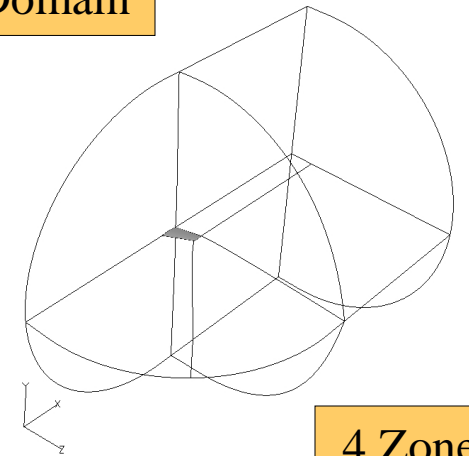
1. Flow Domain, *Extent of CFD analysis*
2. Zone, *Divide domain for convenience, if needed*
3. Grid, *Divides the zone into cells*
4. Cell, *Smallest control volume, but “finite”*



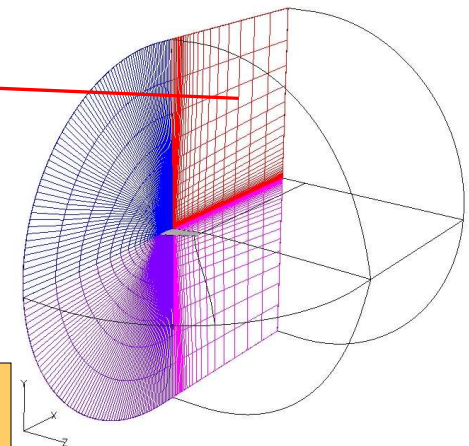
Let's Examine a
hexahedral cell

Each control volume is “air-tight”

1 Domain



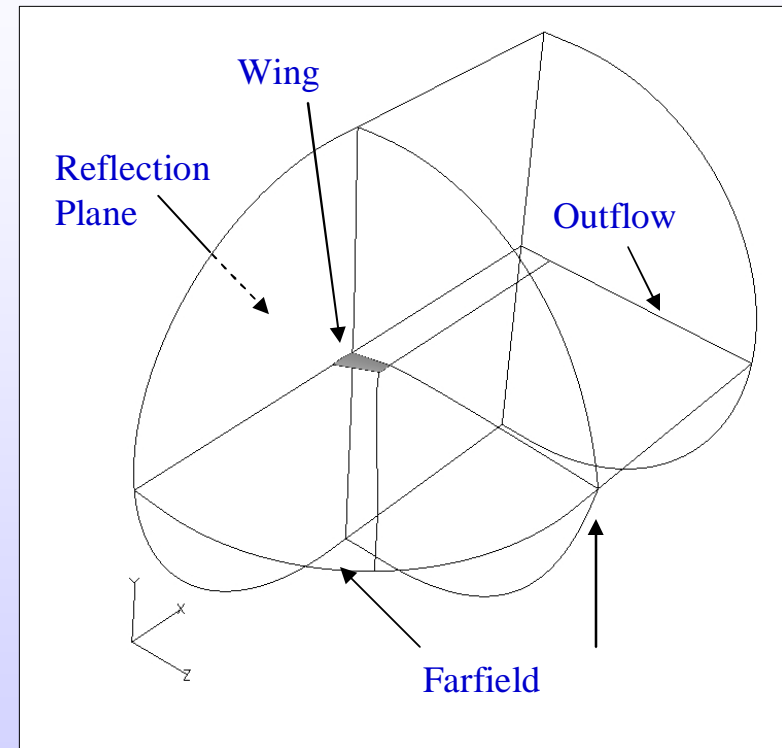
4 Zones



Grid in each zone
with 1000s of cells

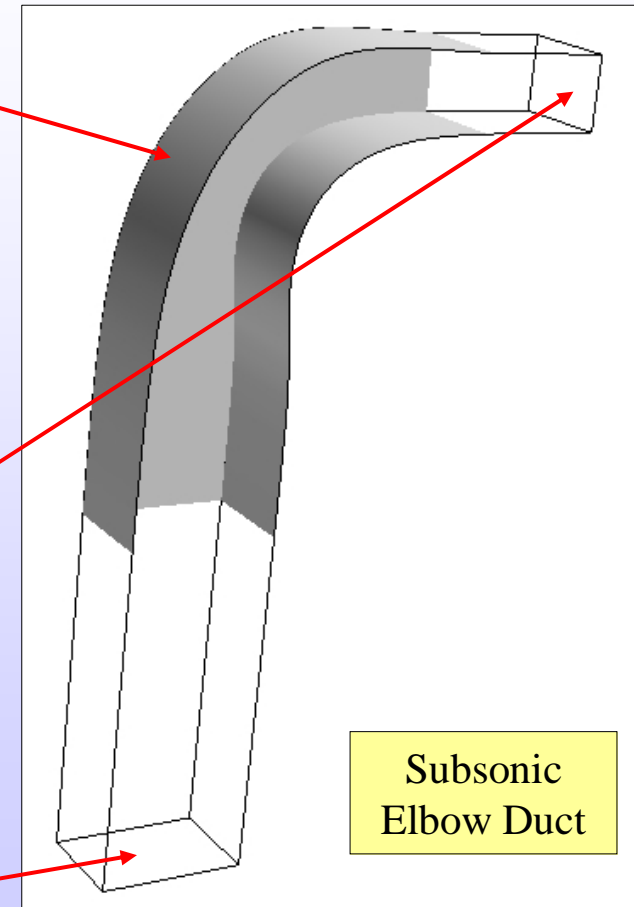
Flow Domain Modeling 1 of 3

- Flow domain defines a closed volume.
- The flow domain requires modeling using similar methods as the geometry modeling.
- Constructing the flow domain usually requires sketching out the topology of the grid (especially for multi-zone, structured grids in which each zone must adhere to a structured topology).
- May need to go to great lengths to fit a structured grid topology onto a geometry. (“clamshells”, singular axis, nose cells).
- Unstructured grids usually allow more flexibility in placing domain boundaries.

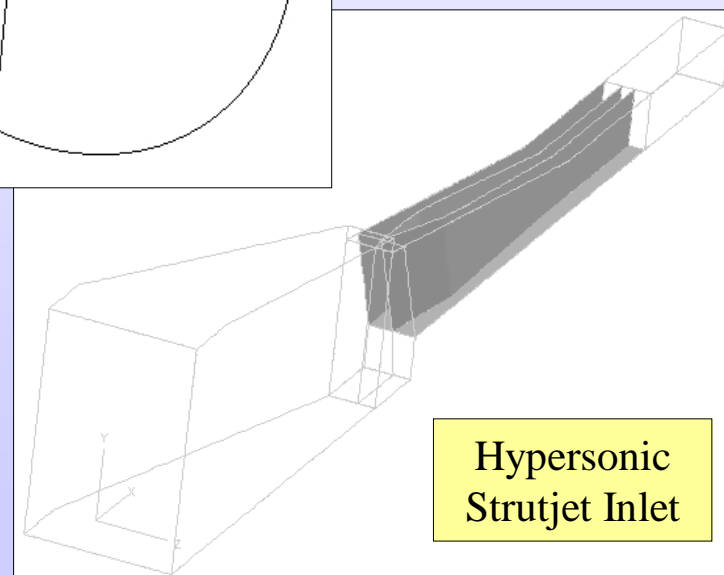
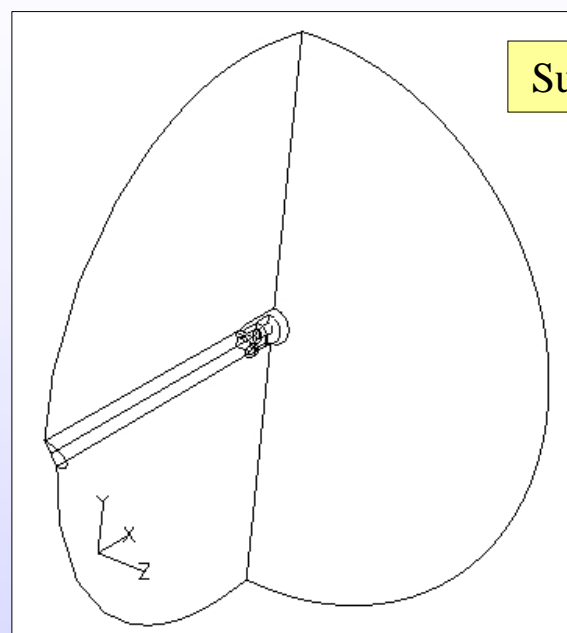
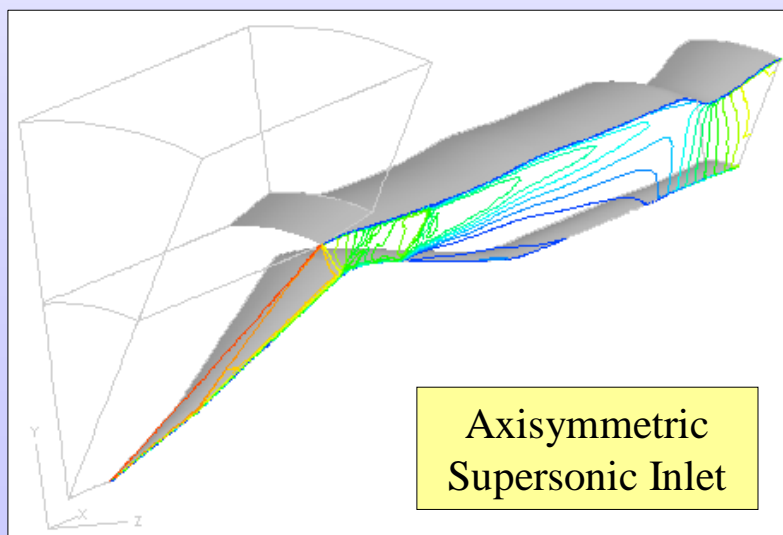


Flow Domain Modeling 2 of 3

- The geometry forms part of the flow domain boundary.
- Useful to know if flow at boundary is subsonic or supersonic and whether it is inflow or outflow.
- Be wary of interactions between inflow / outflow boundaries and the flow field.
- Outflow boundary can be placed downstream to avoid interactions at geometry exit where data is taken.
- Inflow boundary can be placed upstream to create a surface for boundary layer growth.



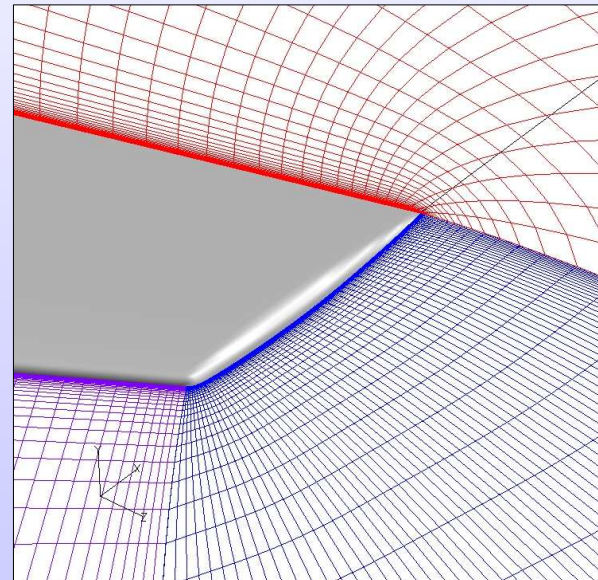
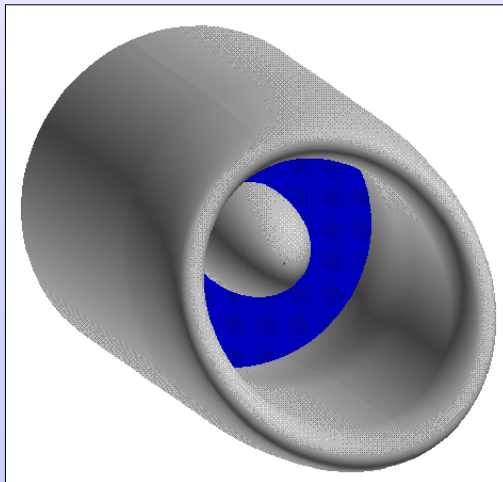
Flow Domain Modeling 3 of 3



Geometry Input to the Grid Generator

The geometry of the body and flow domain are then input to the grid generator for the generation of the grid.

- A surface “database” is used in the grid generator for projecting the surface grid so that the surface grid is true to the geometry.
- Flow domain boundaries are used to construct zonal boundaries.
- Most grid generators can read IGES files.



The grid provides the only geometric information during computation.

General Classification of Grids

Grids can be classified as *Structured* or *Unstructured*:

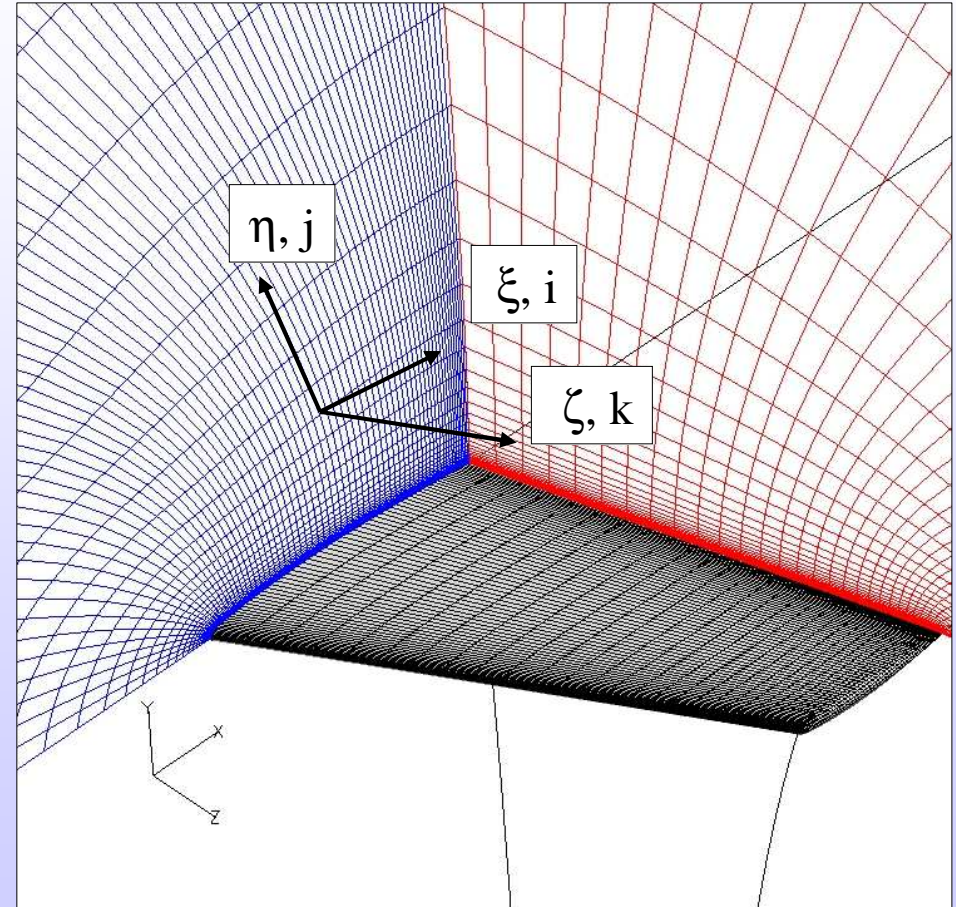
A *Structured Grid* uses a topology in which the cells are arranged in an array structure. Location of neighboring cells is implicit in the array indices (i,j,k). This allows efficient storage and book-keeping of cell information.

An *Unstructured Grid* has no inherent ordering of the cells, and so, the arrangement of cells must be specified explicitly. Unstructured grids allow greater flexibility in generating and adapting grids at the expense of greater storage of cell information.

A *Hybrid Grid* contains both structured and unstructured grids, usually in separate zones.

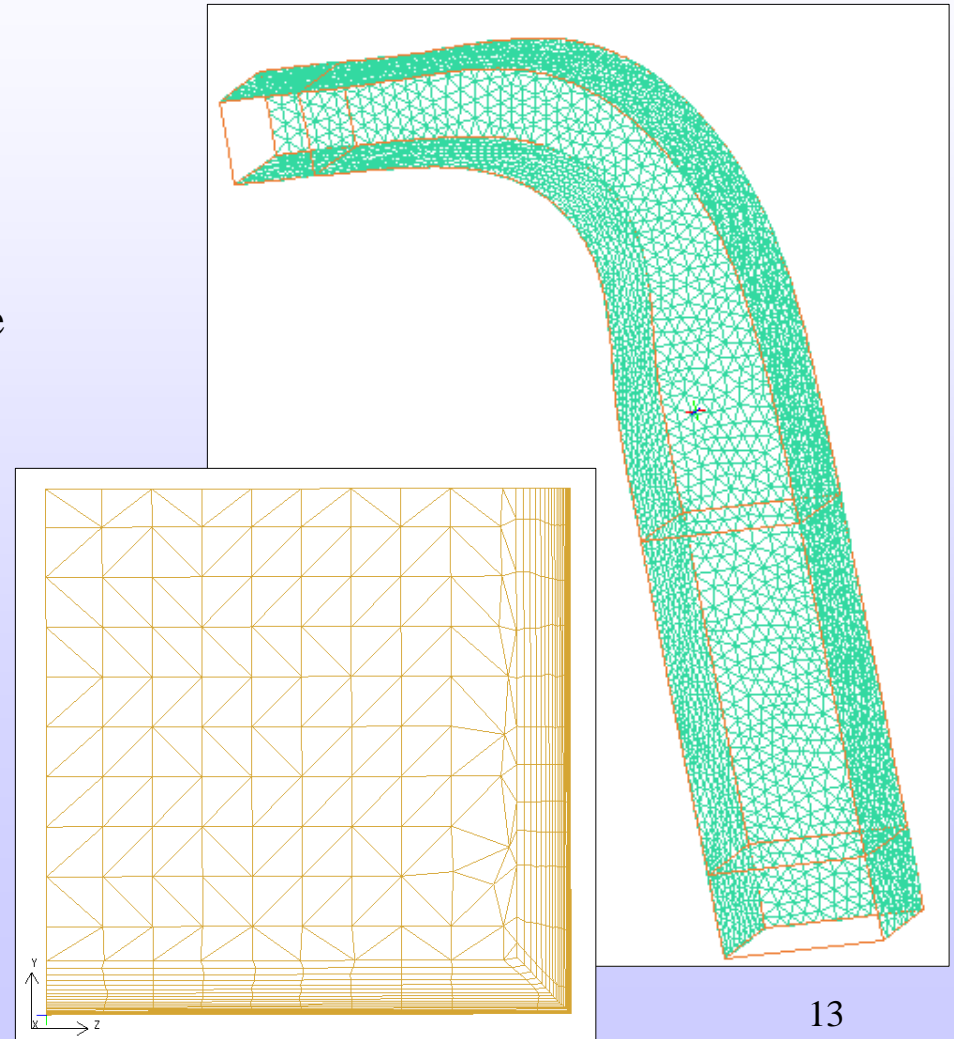
Structured Grid Basics

- Grid is “body-fitted” to follow the shape of the body (wing).
- Grid points (vertices) are arranged in an array structure with indices (i,j,k).
- i-coordinate usually streamwise
- Grids can end up with 10s or 100s of zones and millions of grid points.
- Transformation between physical space (x,y,z) and a Cartesian computational space with coordinates directions (ξ , η , ζ).
- Grid points are clustered to provide resolution of flow gradients.
- Cells are hexahedral (6 quadrilateral sides).



Unstructured Grid Basics

- Grid is “body-fitted” to follow the shape of the body (duct).
- Grid points (vertices) are do not have any set structure.
- No transformation from physical space (x,y,z) and a computational Space.
- Specify and store geometric and connectivity information.
- Grid points are clustered to provide resolution of the gradients.
- Cells are tetrahedral (4 triangular sides).



Integral Equation for Flow

The integral equation for the conservation statement is:

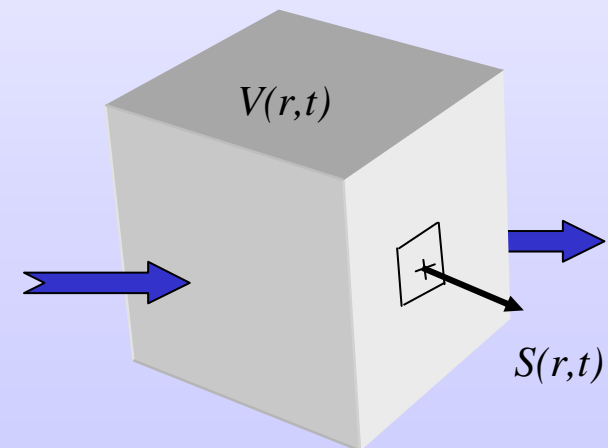
$$\frac{d}{dt} \int_{V(\vec{r},t)} Q dV + \oint_{S(\vec{r},t)} [(\vec{v} - \vec{g})Q - \mathbf{D}] \cdot \hat{n} dS = \int_{V(\vec{r},t)} P dV$$

*Time variation of
Q in volume V*

*Flux of Q through
surface S*

*Production of Q
in volume V*

- Equation applies for a control volume.
- Control surface bounds the control volume.
- Q is conserved quantity representing the flow.
- Flow can be through the control surface.
- Control volume and control surface can vary shape in time and space.
- Flow can be time-varying (unsteady).





Objective of the F-V Formulation

Represent the integral equation as an *ordinary differential equation* (then eventually an *algebraic equation*) amenable to a solution using computational (numerical) methods.

Thus, we need to approximate the *volume integrals* and the *surface integrals* to form algebraic expressions.

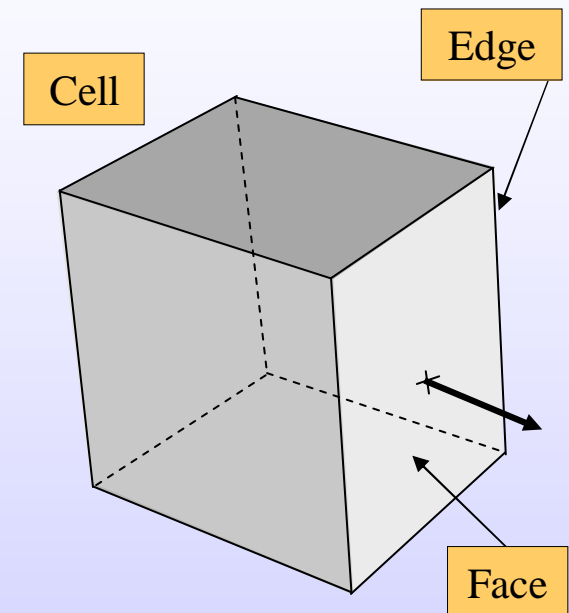
Prior to discussing these approximations, let's examine the control volumes (grids) on which the integrals will be approximated...

Anatomy of a Finite-Volume Cell

Finite-Volume Cell:

- Cell can take on a generalized shape.
- Cell contains a finite (positive) volume.
- Integral equation will be approximated on the cell to form an algebraic relation.
- Size of cell indicates the level of computational resolution of the CFD analysis.
- Control surface is *faceted* into a finite number of **faces**.
- Faces can take on a variety of shapes.
- Face is bounded by **edges**.
- Edges are usually straight lines.

As the cells become smaller and the number of cells in a grid increases, the computational effort increases.



Hexahedral Cell
6 Quadrilateral Faces
12 Linear Edges
(4 per face, edges shared)

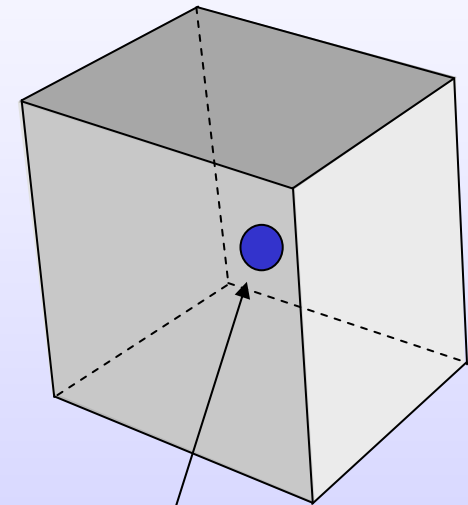
Volume Integral Approximation

Approximation: *Q is uniform within the finite-volume:*

$$\int_{V(\vec{r},t)} Q dV \approx V Q = \hat{Q}$$

$$\int_{V(\vec{r},t)} P dV \approx V P = \hat{P}$$

*The position of the solution point
in the cell is not yet defined.*



$$V_i Q_i = \hat{Q}_i$$

$$V_i P_i = \hat{P}_i$$

Cell i
(i is an index for the cell)

Surface Integral Approximation

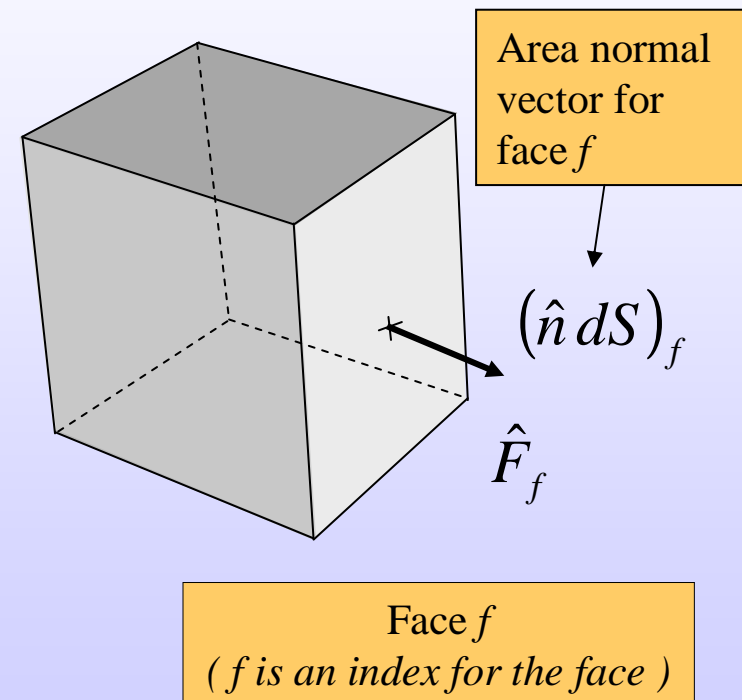
Approximation: *The flux is uniform over the surface of each face of the cell:*

$$\oint_{S(\vec{r},t)} [(\vec{v} - \vec{g})Q - \mathbf{D}] \cdot \hat{n} dS \approx \sum_{f=1}^{nf} \hat{F}_f = \hat{F}$$

where the flux on the face is define as

$$\hat{F}_f = [(\vec{v} - \vec{g})Q - \mathbf{D}]_f \cdot (\hat{n} dS)_f$$

Computing the flux on the face is one of the most difficult and computationally intensive operations of a CFD code.



Integral Equation

Start with the integral equation,

$$\frac{d}{dt} \int_{V(\vec{r},t)} Q dV + \oint_{S(\vec{r},t)} [(\vec{v} - \vec{g})Q - \mathbf{D}] \cdot \hat{n} dS = \int_{V(\vec{r},t)} P dV$$

and substitute in the volume and surface integral approximations to yield:

$$\frac{d\hat{Q}}{dt} = \hat{P} - \hat{F}$$

This equation is a first-order, non-linear, ordinary differential equation for which various numerical methods exist for its solution.

A Further Simplified Form

Further define

$$\hat{R} = \hat{P} - \hat{F}$$

which will result in the form

$$\frac{d\hat{Q}}{dt} = \hat{R}$$

This form will be used to simplify the discussion of the time-marching methods.

Other Cell Shapes

A *structured grid* can only contain finite-volume cells with a *hexahedral* shape. *Unstructured grids* allow greater freedom for cell shapes. Possibilities include:

- Generalized Cell (*X quadrilateral faces, Y triangular faces*)
- Prismatic Cell (*3 quadrilateral faces, 2 triangular faces*)
- Pyramidal Cell (*1 quadrilateral face, 4 triangular faces*)
- Tetrahedral Cell (*4 triangular faces*)

To keep cell geometry simple, *quadrilateral* or *triangular* faces with *straight-line* edges are generally used. The geometry and the normal area vector of a triangle is uniquely known, and so, quadrilaterals are usually divided into triangles to compute their geometric properties.



Location of the Solution in the Cell

The location of the flow solution and geometry of the finite volume cell with respect to the grid can be of two types:

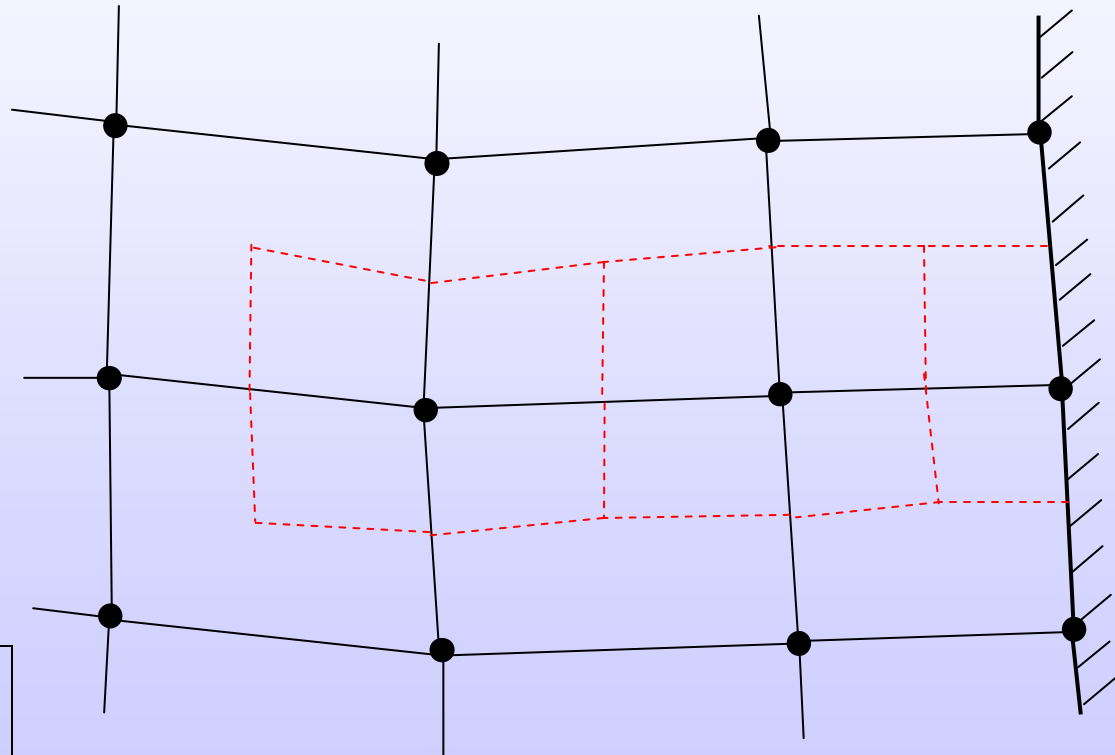
Cell-Vertex (Node-Centered) Cell. The flow solution is located at the vertices of the grid. The finite-volume cell is formed about the vertex.

Cell-Centered Cell. The flow solution is located at the centroid of the cell volume defined by the grid lines (primary grid).

Each approach has its advantages and disadvantages, but if things are done right, both approaches do well.

Cell-Vertex Cell

- Solution located at vertices.
- Cell formed about vertex.
- Half-cell at the boundary.
- Solution point at boundary.



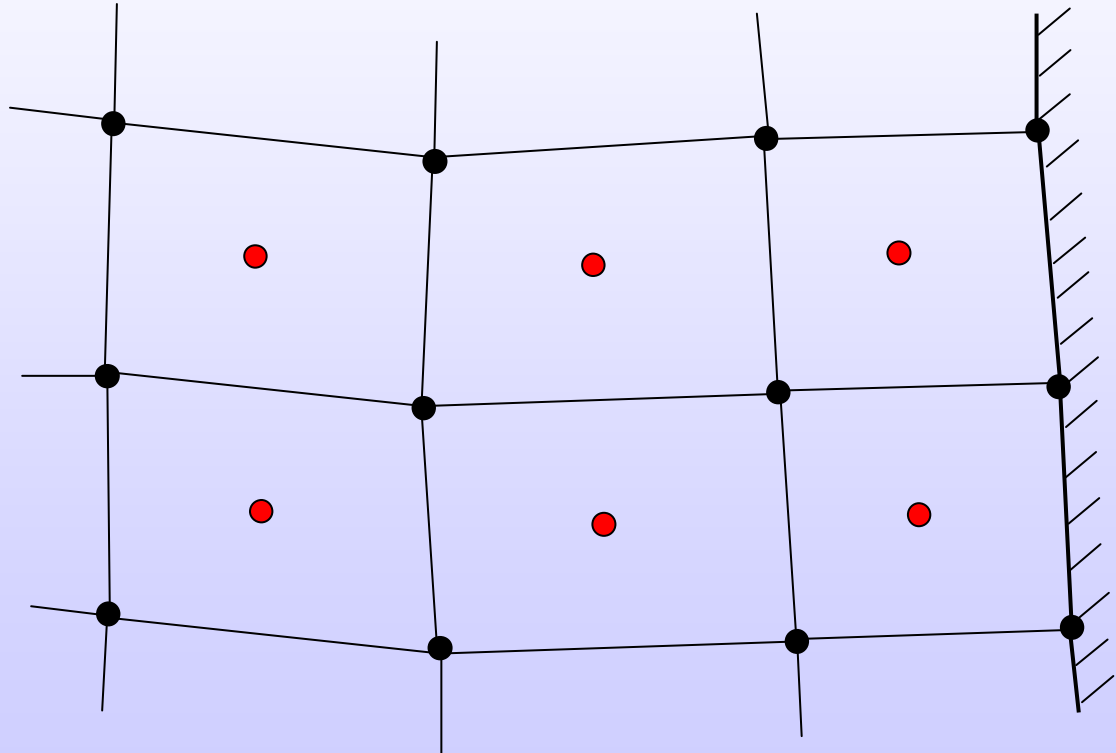
Vertex / Solution	●
Grid lines	—
Boundary	—
Cell edges	- - -

WIND uses a cell-vertex finite-volume cell.

Cell-Centered Cell

- Solution located at centroid.
- Grid forms the cell.
- Full cell at the boundary.
- Flux at boundary.

Vertex	●
Solution	●
Grid / Edge	—
Boundary	—





Simplified Cell Shapes

Often assumptions can be applied to simplify the geometry of the flow domain, grid, and cells from a three-dimensional geometry:

Quasi-three dimensional cell. Grid is planar (x,y) with the z -coordinate varying to indicate variable depth of the cell.

Planar axisymmetric cell. Grid is planar (x,y) with y indicating the distance from an axis-of-symmetry. Angle of axisymmetric wedge indicates depth.

Planar two-dimensional cell. Grid is planar (x,y) with the z -coordinate indicating the fixed depth of the cell.

Quasi-one-dimensional cell. Grid is one-dimensional (x) with the cross-sectional area variable and specified along x .

One-dimensional cell. Grid is one-dimensional (x) with the cross-sectional area constant along x .

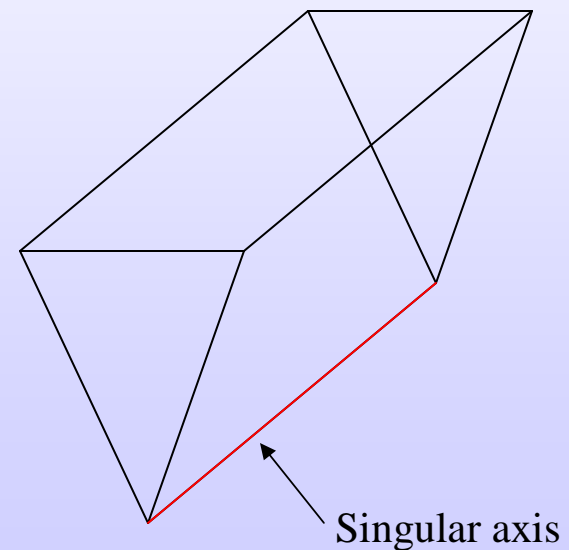
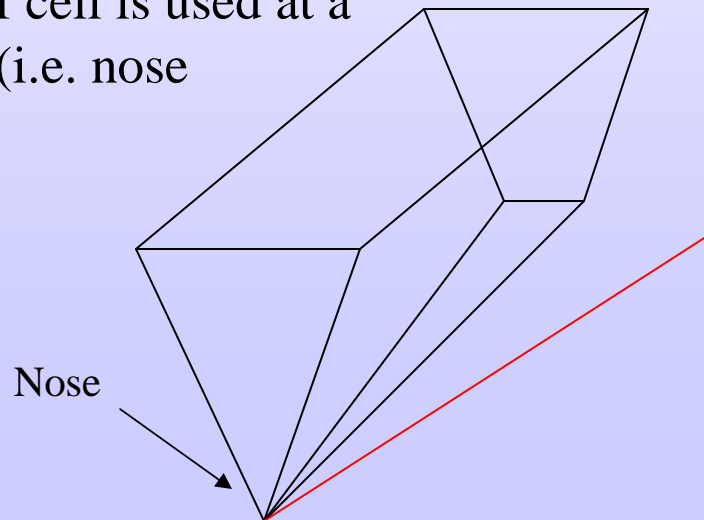
WIND needs at least a 2D grid (x,y) .

Degenerate Cell Shapes

Degenerate cell shapes are sometimes used to build in flexibility:

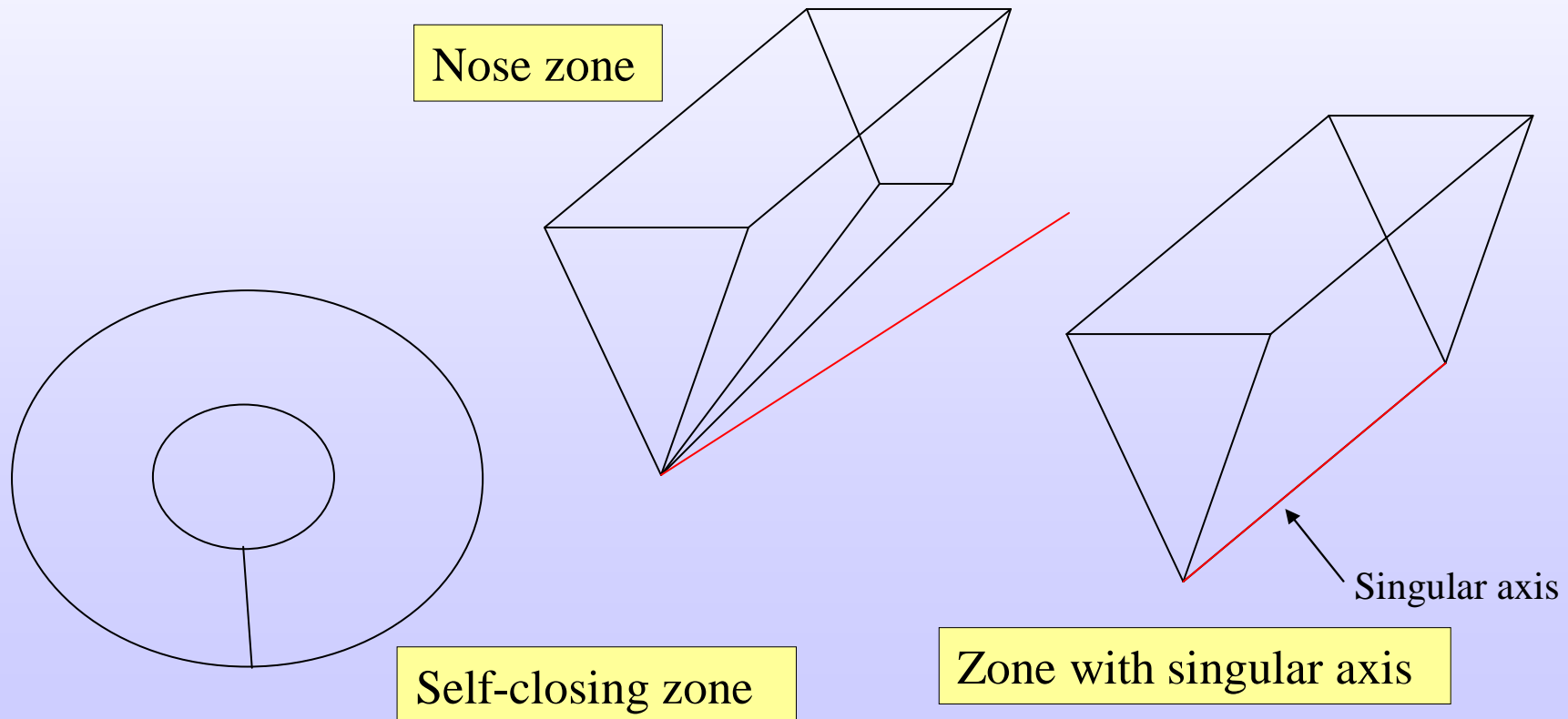
Wedge-shaped cell. The hexahedral has one face on the boundary that has collapsed to a line that is a singular axis. Since face has zero area, the flux is zero, so all is fine with the numerical methods. Special boundary condition is usually applied to handle these.

Sharp Nose Cell. The hexahedral has one edge that has collapsed to a point. This type of cell is used at a sharp nose (i.e. nose of a cone).



Single-Zone Topology

A zonal boundary of a zone may be topologically connected to itself or other boundaries of the same zone:



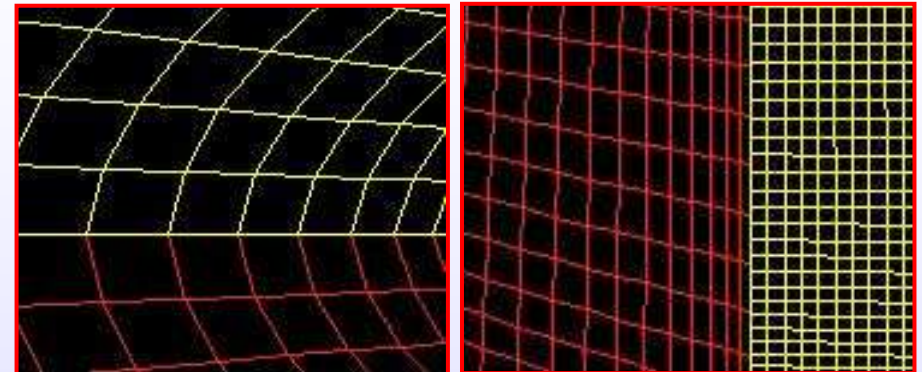
Multi-Zone Connectivity

- Flow information is exchanged across zonal boundaries.
- Connectivity defines how a zonal boundary is connected to other zonal boundaries.

Types of Zonal Connectivity

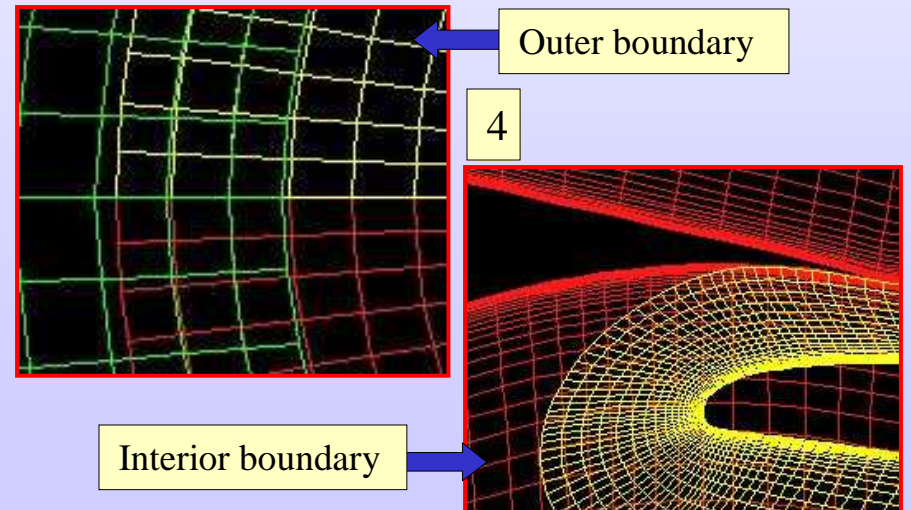
- 1) Abutting, Point-to-point match
- 2) Abutting, Non point-to-point match
- 3) Overlapping, Point-to-point match
- 4) Overlapping, Non point-to-point match

- GMAN can automatically compute zonal connectivity for abutting (1 & 2).
- Overlapping (overset) grids use tri-linear interpolation to exchange information. This works best when cells in overset region are approximately the same size.



1 & 3

2

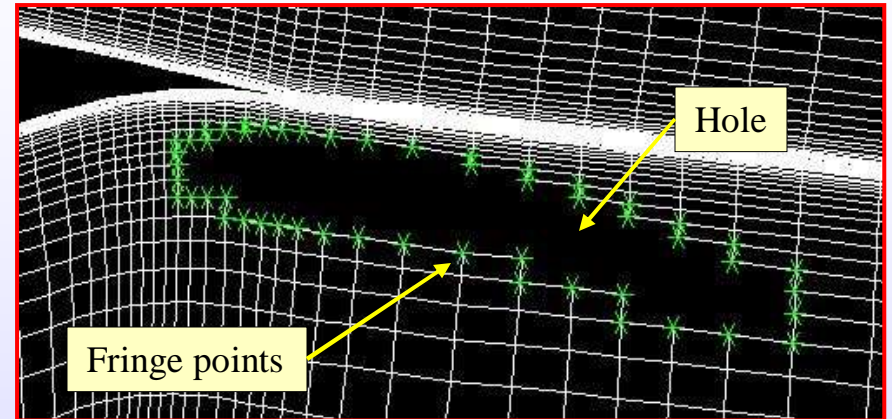
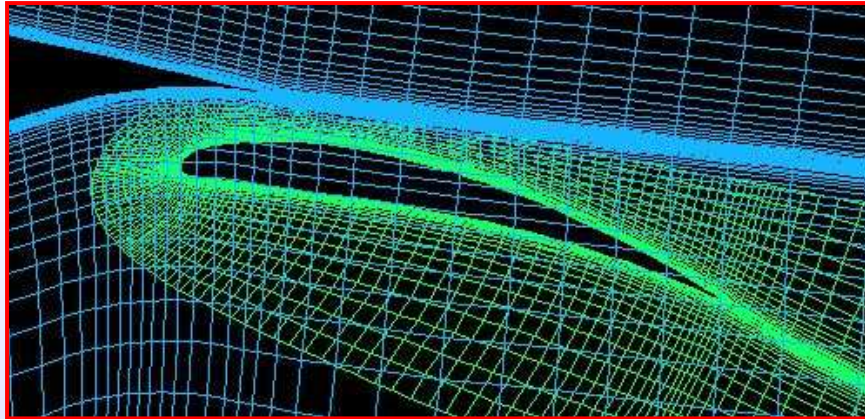


Outer boundary

4

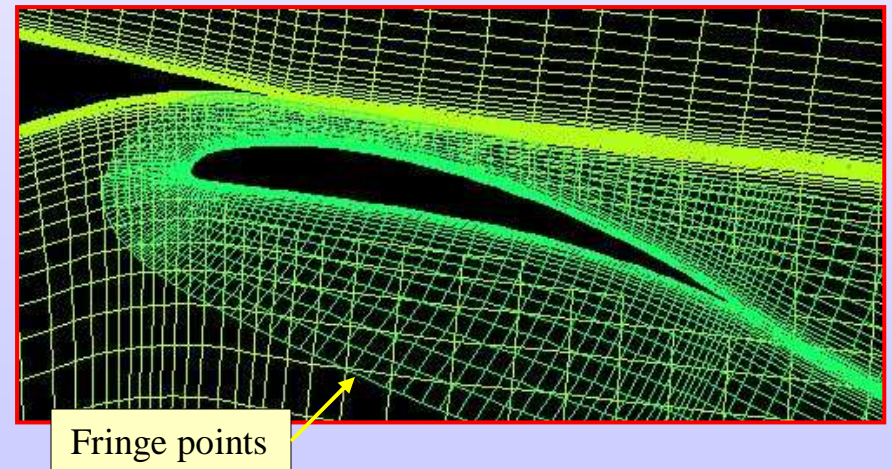
Interior boundary

Overlapped Grids



Example of NLR Airfoil with flap

- Flap grid overlaps airfoil grid.
- Use flap grid to cut a hole in airfoil grid.
- Indicate edge points of cut are fringe points to interpolate boundary information from flap grid solution.
- Outer edge of flap grid is fringe boundary of flap grid and receives interpolated data from the airfoil grid.





Remarks on Zonal Connectivity

- Avoid placing zonal boundaries in regions with large flow gradients (shocks, boundary layers, shock / boundary layer interaction). *This is often unavoidable.*
- Best if zonal boundaries are place *normal* to the flow direction.
- Placing zonal boundaries parallel to the flow in regions of high gradients is generally not good, but often unavoidable.
- Point-to-point matching across zones is best since inviscid flux information can be directly transferred and errors are less in interpolation of viscous and turbulent information.
- Best if zones overlap by at least two grid points, especially in turbulent flows. This improves the interpolation. However, this may be difficult for grids with lots of zones and complicated geometry.
- Tools are available for combining, splitting, and overlapping zones in a grid.



Grid Quality Parameters

Grid quality parameters can be defined that can be used in the grid generation process:

- Grid spacing normal to a surface
- Grid spacing (minimum, maximum) along curves and surfaces
- Maximum grid spacing within the zone
- Maximum grid stretching
- Match of grid spacing across zonal boundaries
- Grid orthogonality

Grid Clustering at Walls

Walls with boundary layers, require the grid to be “clustered” normal to the wall to resolve the large gradients through the boundary layer.

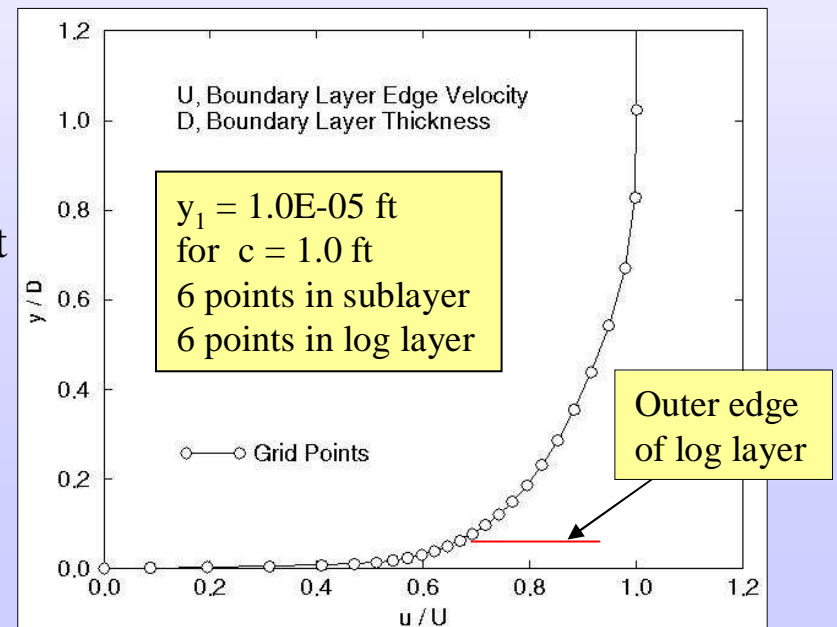
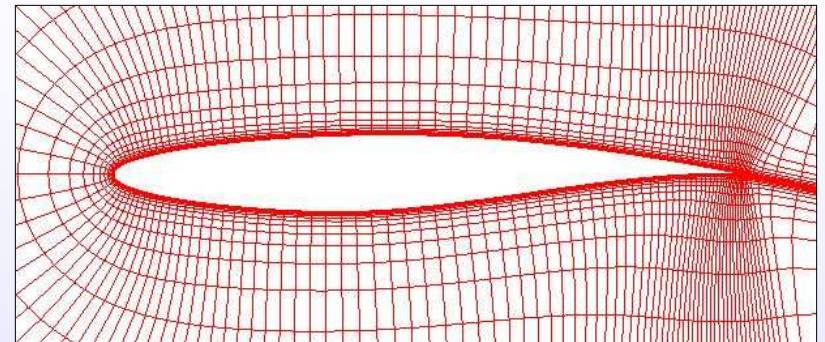
Non-dimensional distance from a wall in a turbulent boundary layer is given by y^+

$$y^+ = y \frac{\sqrt{\rho_w \tau_w}}{\mu_w}$$

General rule is that first grid point off the wall should have $y^+ < 1$ to accurately define a turbulent velocity profile. Can go up to $y^+ = 5$ for less accuracy. Resolution of heat fluxes requires y^+ about 0.1.

With a smooth grid about 5-10 grid points will be in the viscous sublayer ($y^+ < 30$).

Even inviscid wall needs a little clustering.

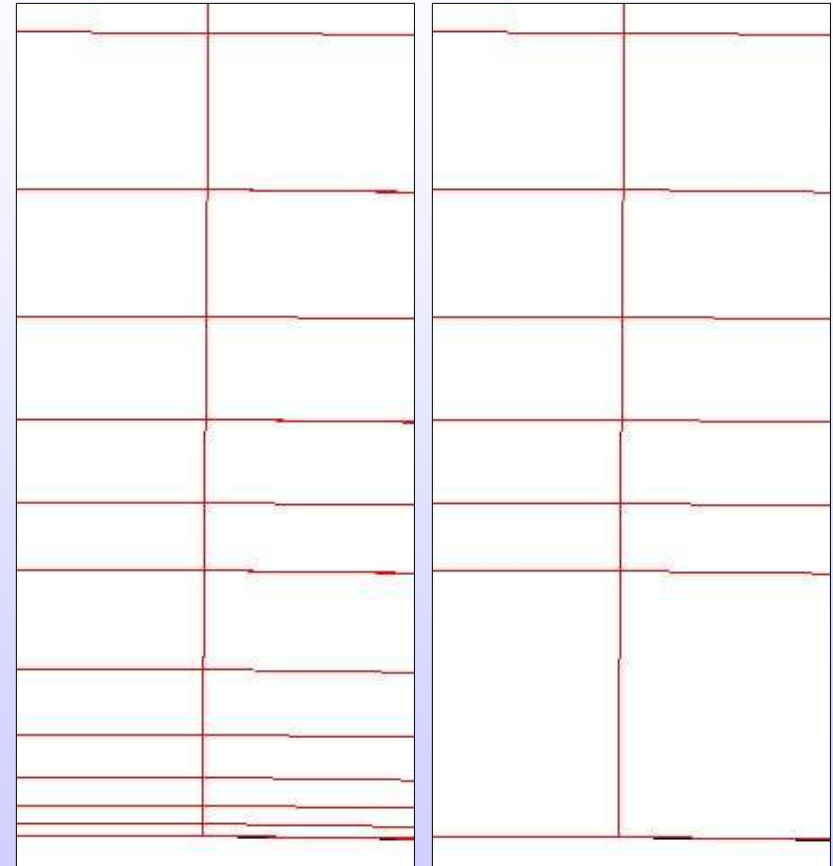


Grids for Wall Functions

A wall function computes properties in the log layer ($15 < y^+ < 100$) of an attached turbulent boundary layer at a wall.

- Can remove the grid points in the log layer.
- This reduces the overall number of grid points.
- Also removes the smallest grid cells that inhibit iterative convergence.

The full grid is generated and then the log region points are removed (CFSUBSET utility does this).



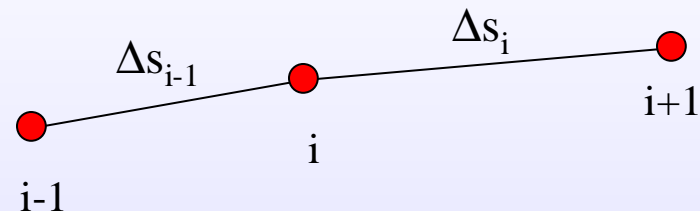
Full grid.

Grid for use with
a wall function.

Grid Stretching and Smoothness

Grid stretching is the ratio of distances between adjacent grid points along a grid line.

$$S = \frac{\max(\Delta s_i, \Delta s_{i-1})}{\min(\Delta s_i, \Delta s_{i-1})}$$



Stretching is defined so that $S \geq 1$.

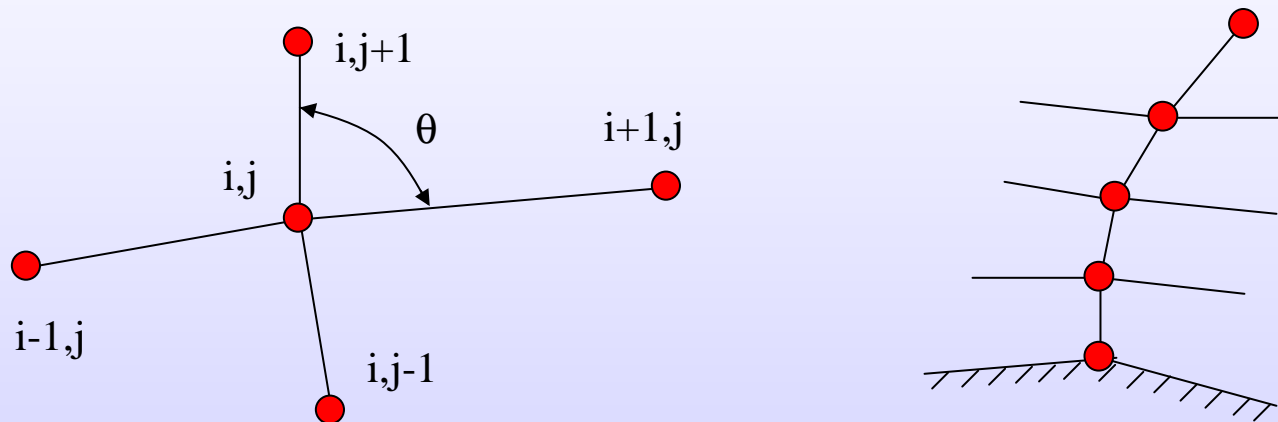
Can express as a percentage as $S\% = (S-1)*100\%$

General rule is to keep stretching **below 15-20%** to keep numerical errors within bounds. This is especially important in regions with strong flow gradients. Stretching upwards of 30% to 35% could be used for simulations where accuracy demands are less.

A **smooth grid** is one in which the stretching varies smoothly over the grid.

Grid Orthogonality

Grid orthogonality is the angle that a grid line makes with the other grid lines intersecting at a grid point.



Orthogonality is defined so that $\theta \leq 90^\circ$.

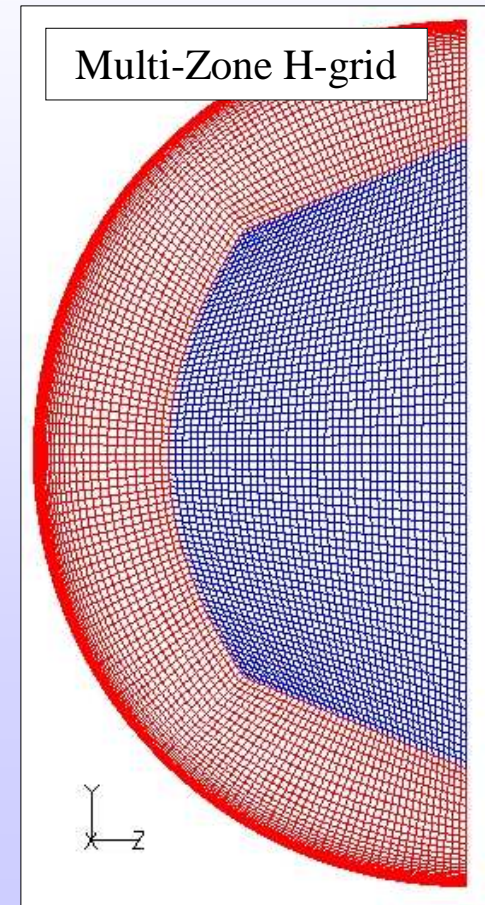
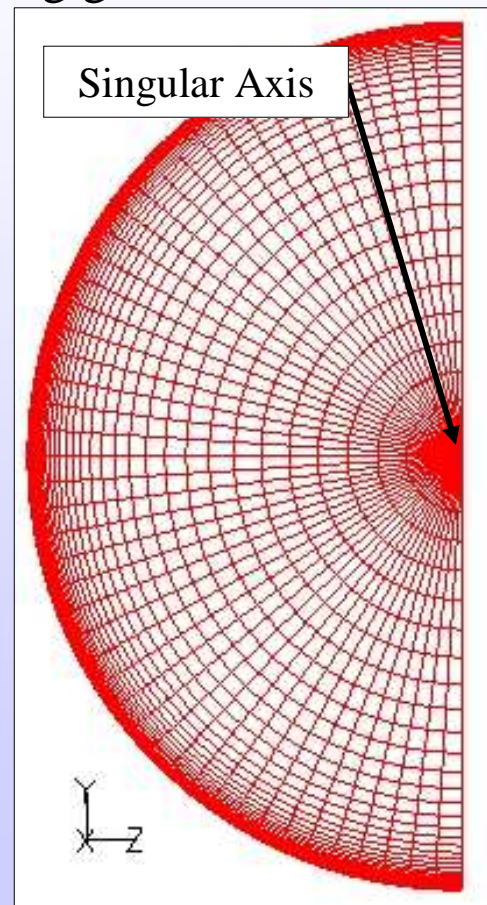
General rule is to keep orthogonality as close to 90° as possible, especially normal to solid surfaces to provide accurate estimates of the normal to the walls for wall boundary conditions. A minimum angle is about 45° .

Duct Grids 1 of 2

Several options exist for generating grids in ducts

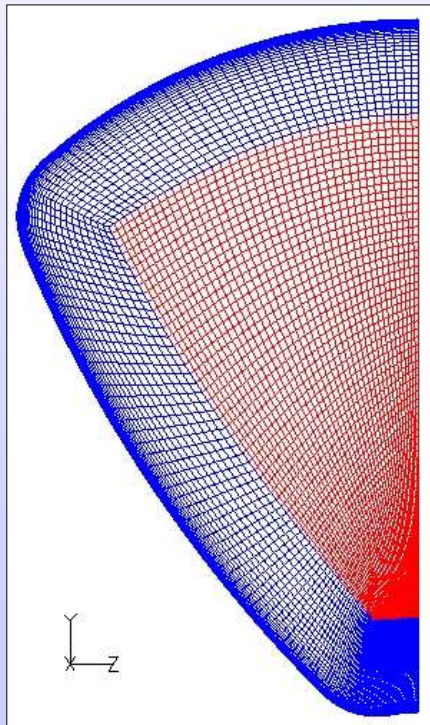
Constant- i grid surfaces

1. Singular axis
2. Single-Zone H-grid
3. Multi-Zone H-grid



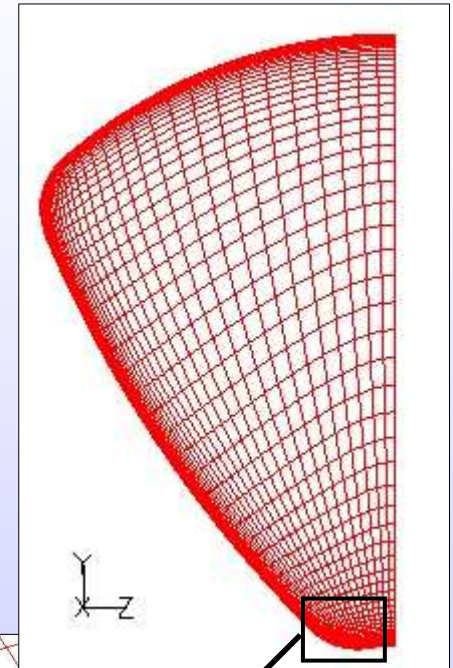
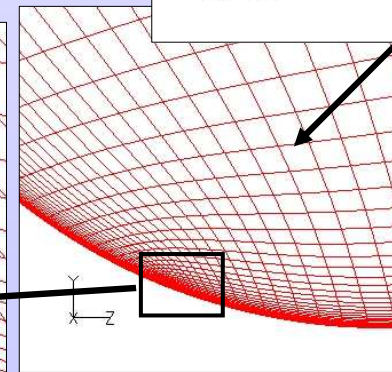
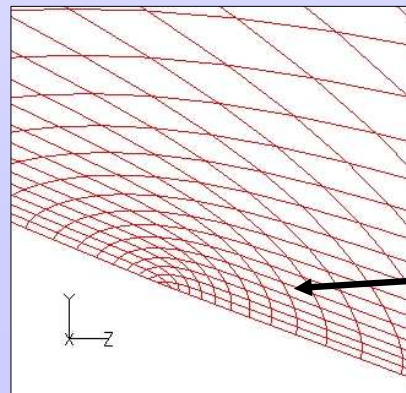
Duct Grids 1 of 2

Two zones: Blue zone wraps about inlet surface. One face of blue zone matches to three faces of red zone.



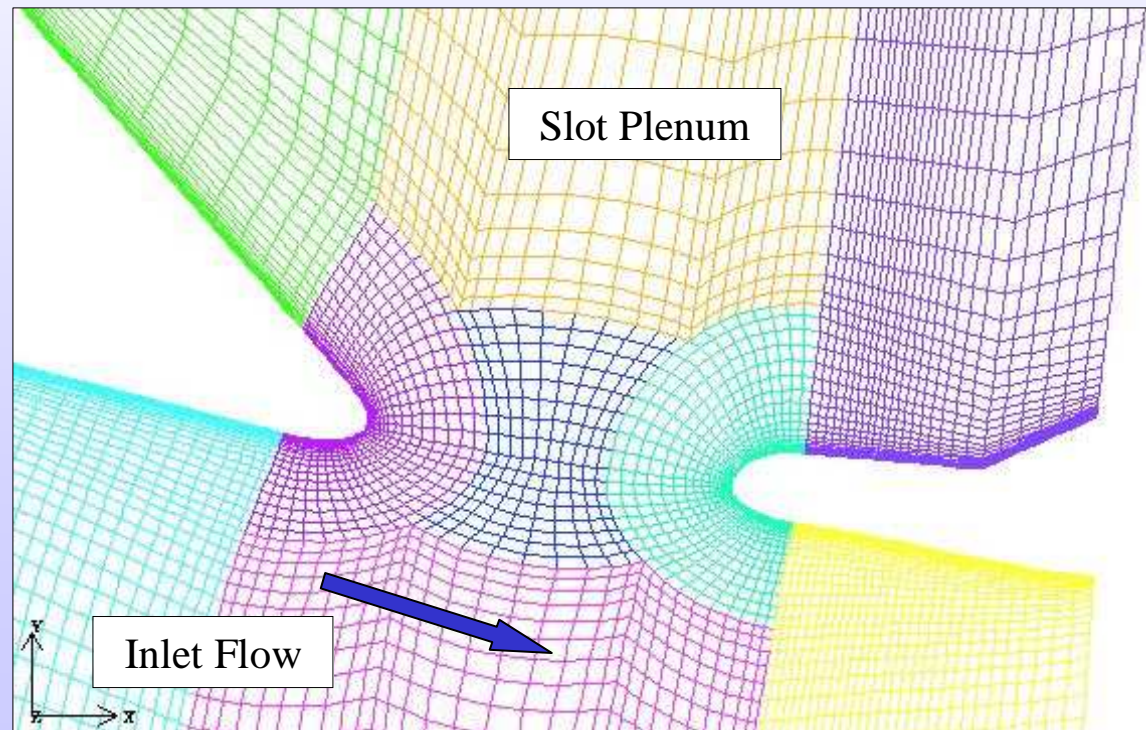
H-grid for a circular corner

- Grid generator needs to generate orthogonal grids at the boundary
- Gridgen has been used with success. Use elliptic for “domain” grids and the TFI for “volume” grids.
- Cells will have positive volume



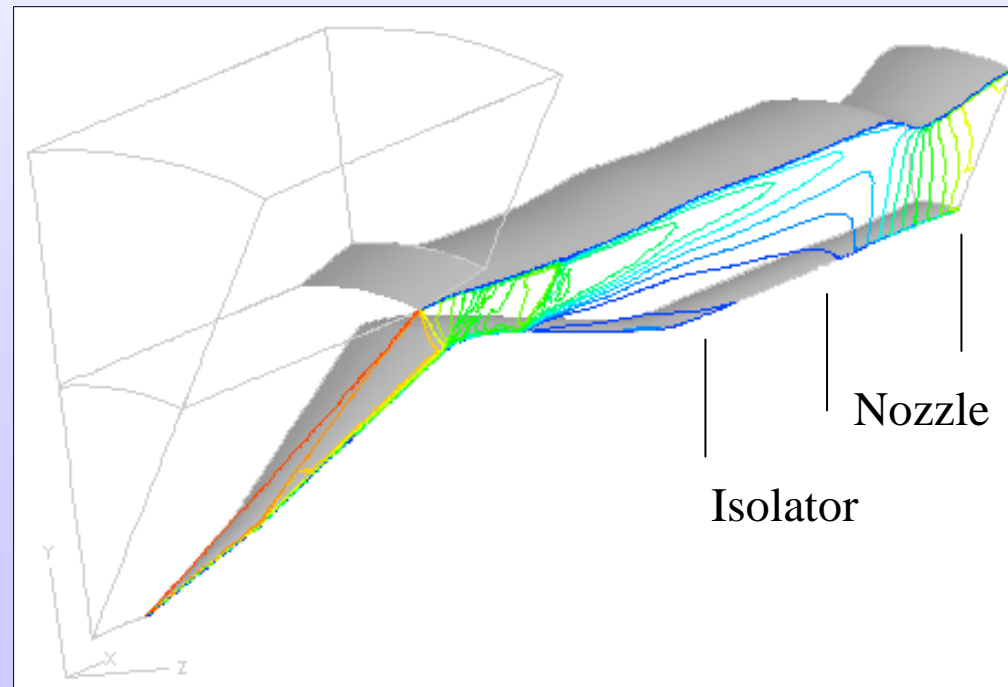
H-C Slot Grid

- Region consists of 9 zones
- Grid clustering on walls wraps around the lips of the slot
- Grid lines match across the zonal boundaries



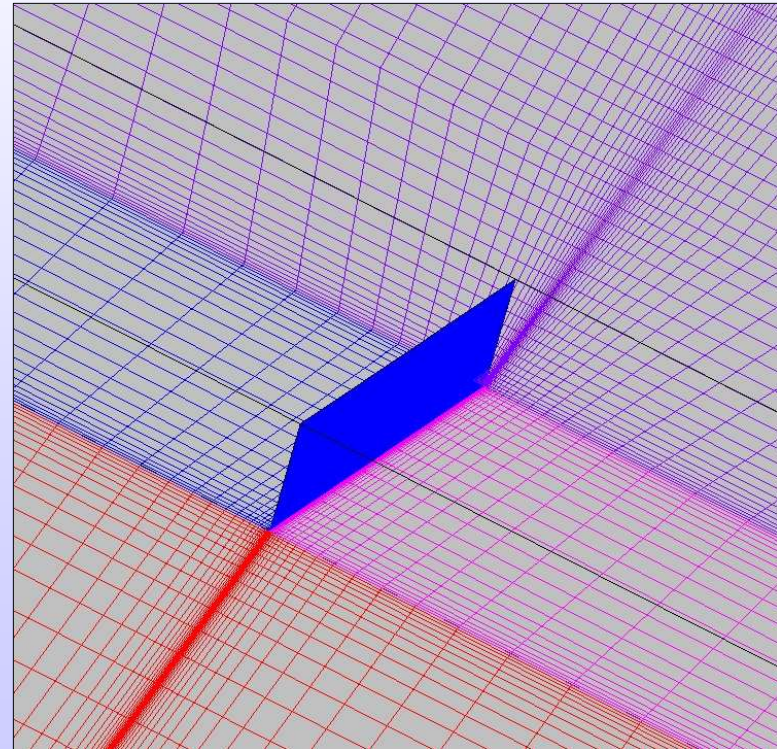
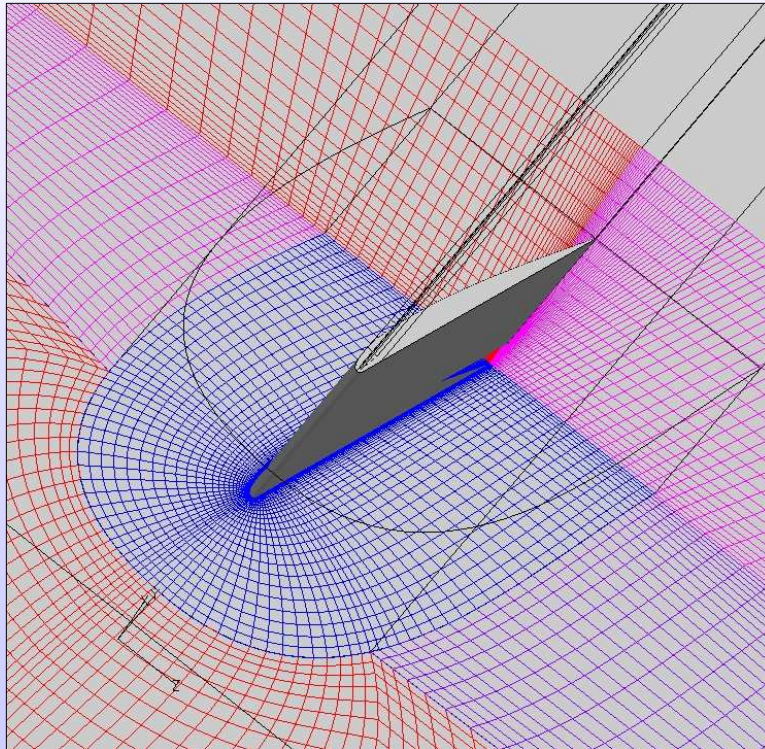
Nozzle Outflow Zone

Setting the outflow mass flow of an inlet can be done by attaching a converging-diverging nozzle at the outflow plane and varying the nozzle radius. The flow through the nozzle throat becomes choked and the outflow boundary condition at the nozzle exit is a simple extrapolation of supersonic flow.



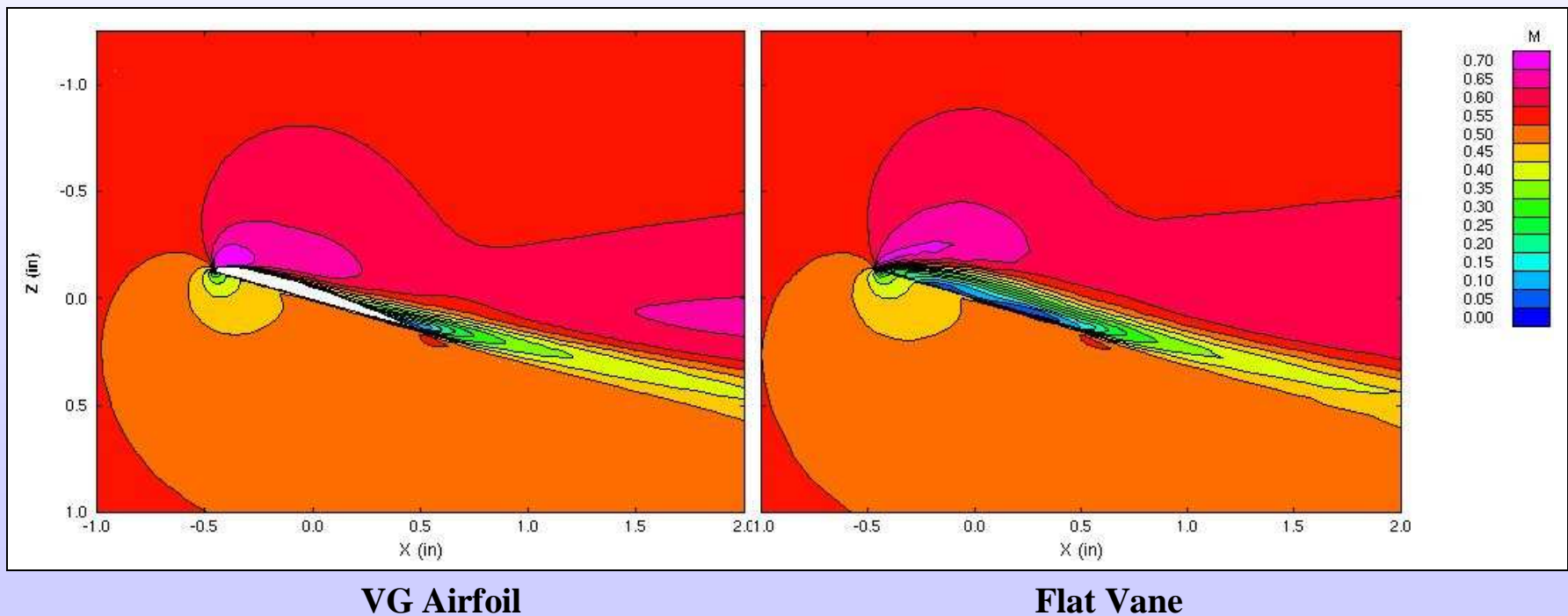
Gridding Vortex Generators 1 of 3

Gridding up a VG can introduce large number of grid points.
Simplifying alternative is to assume a flat shape.



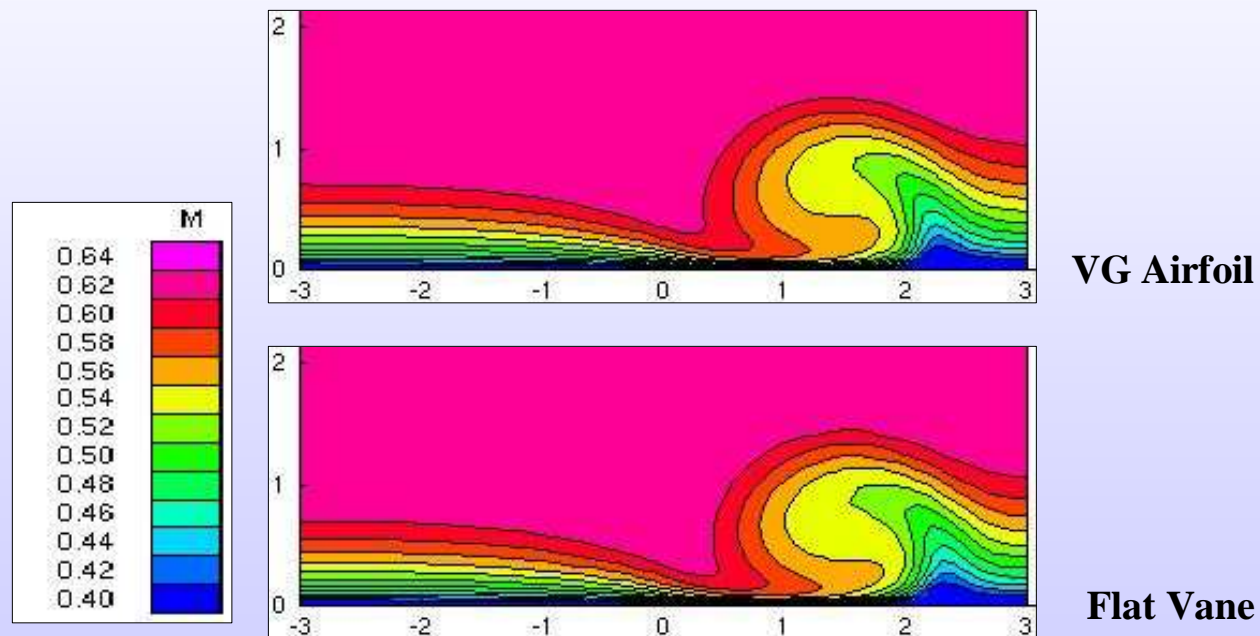
Gridding Vortex Generators 2 of 3

Flowfields at airfoil and flat vane surfaces are different, but yet similar away from the VG.



Gridding Vortex Generators 3 of 3

Flow downstream is similar.





Grid Sequencing

Grid sequencing in a CFD solver is when the solution is performed on only certain grid points, such as only every other grid point along each grid line. This essentially coarsens the grid.

- Fewer grid points reduces the number of computations, and so, CPU time.
- Larger grid spacings help damp higher frequencies that limit iterative convergence.

This is a good approach for starting the solution, damp out initial transients, and get a coarse-grid solution for a grid convergence study.

The requirement for grid generation is that the number of grid points along a grid line N be constrained by

$$N = 2^n m + 1$$

where n is the number of levels for sequencing and m is some integer. One level of sequencing would mean every other grid point is solved, and so, require $n = 1$.



CAD and Grid Generation Tools

Some CAD Tools:

- AutoCad
- Pro/Engineer
- Unigraphics
- ICEM DDN

Some Grid Generation Tools:

- GRIDGEN
- ICEM CFD
- NPARC System provides tools for scaling, rotating, manipulating, translating grids and interpolating solutions onto new grids.
- NASA Langley has number of grid tools (VGRID, GeoLab)
- Variety of smaller, specialized codes